# Low Leakage SRAM design using sleep transistor stack

Abhishek Mathur, Arun Jayachandran, Ramya Venumbaka

***Abstract*— Low power design is the industry buzzword in present chip design technologies. Caches occupy around 50% of the total chip area and consume considerable amount of power. This project's focus is to reduce leakage power consumption of an 8kbit SRAM by employing techniques like power gating. The main technique used in power gating is the use of sleep transistor. In our design we have chosen a stack-based implementation.**

*Index terms* – **Low-power SRAM, leakage savings, sleepy stack, MTCMOS.**

## I. Introduction:

CMOS technology scaling continues to reduce switching delay and power while improving area density. But transistor miniaturization brings along with it challenging issues like process variations and increase in transistor leakage. Modern high performance microprocessors show increase in leakage as technology is scaled further. A high performance VLSI microprocessor demands huge SRAM clusters to meet performance requirements. There are three components of the leakage power; sub threshold leakage, gate leakage and junction leakage. Lowering power supply or dropping the rail-to-rail voltage decreases the leakage power. By this voltage scaling, the SRAM cell stability is degraded. And also the SRAMs are required to retain the data. Hence, the rail-to-rail voltage needs to be carefully controlled to maintain cell stability, to avoid data loss [2].

As we can see from the figures below, the power leakage is getting worse with scaling [3]. Additionally since SRAM component of the design contributes much to the total chip area, their leakage power has become a significant component of total chip power.

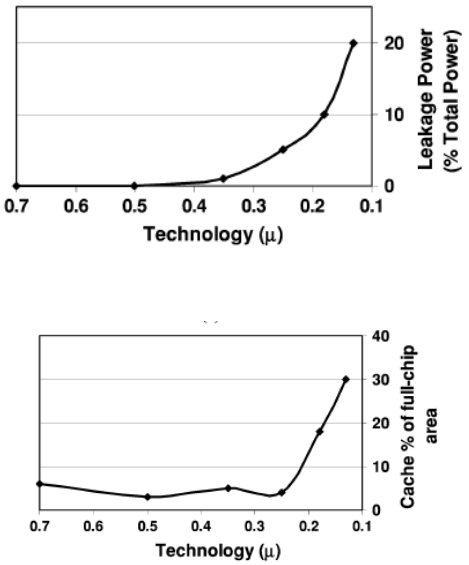


Fig. 1

Power Gating:

To reduce the leakage power, the power gating method can be applied. Power gating is the technique of disconnecting the components in the circuit temporarily that are currently not in use. A very basic implementation of power gating is to use an externally switched power supply and shut-off supply when required. CMOS switches that provide power to the circuit can be controlled to do so. A major technique of power gating is to use sleep transistors to control the sub-threshold current [8]. In this project, we have adopted a sleepy structure that merges stack technique and sleep transistor. Normal SRAM cells have lower threshold voltages but the sleep transistors have higher threshold voltages. Typically low leakage PMOS transistors are used as header switches to shut-off power supply. Footer NMOS devices are also used to control power supply to the circuit.

## II. Design and Implementation

### a) SRAM Cell – 6T:

Bit 0 or 1 in a SRAM cell is stored using two cross coupled inverters. This storage cell has two stable states **0** and **1** which is reinforced because of cross coupling. Two additional *access* transistors serve to control the access to the storage cell during read and write operations. So a typical SRAM cell is a six transistor structure. A 6T SRAM cell requires a careful device sizing to ensure read stability, write margin and data retention in standby modes. The figure below shows a typical 6T SRAM cell and the table shows the corresponding transistor sizes. Access to the cell is enabled by the word line which controls the two access transistors M5 and M6. They in turn control whether the cell should be connected to the bit lines. Bit lines are used for both read and write operations. Two bit lines are not necessary but they are provided to improve noise margins. In read stability, M1 transistor is required to be much larger than M5 transistor to make sure that the node between M1 and M5 does not flip. In write mode, bit lines overpower cell with a new value. High bit lines must not overpower inverters during read operation. So, M2 is designed to be weaker than M5.

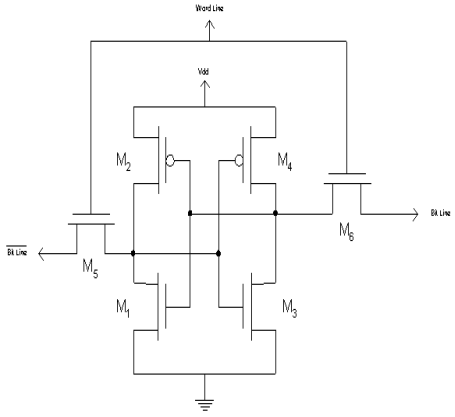


Fig 2. SRAM Cell

| Transistor | W/L (nm) |
|---|---|
| M1 | 500/120 |
| M2 | 190/120 |
| M3 | 500/120 |
| M4 | 190/120 |
| M5 | 250/120 |
| M6 | 250/120 |

Table.1

### b) SRAM Functionality:

A SRAM cell has three different modes of operation [1]:

a) Standby:

If the word line is not asserted, the access transistors M5 and M6 disconnect the cell from the bit lines. The two cross coupled inverters formed by M1-M4 continue to reinforce each other as long as they are disconnected from the outside world.

b) Read:

The read cycle starts by pre-charging both the bit lines to a logical 1 and then asserting the word line, enabling both the access transistors. If a 1 is stored in the cell, this value is transferred to the bit lines by leaving BL (bit line) at its pre-charged value and discharging $\underline{BL}$ to a logical 0 through M1 and M5. The transistors M4 and M6 pull the bit line to a logical 1. If the content of the memory is a 0, then BL is pulled to a logical 0 and BL to a logical 1.

c) Write:

If a 0 is to be written, BL and $\overline{BL}$ are set to 0 and 1 respectively. A 1 is written by inverting the values of the bit lines. WL (word line) is then asserted and the value that is to be stored is latched in.

### c) Sleep Transistors

Sleep transistors are generally high $V_t$. They are switched off when idle and can help save about 40% leakage power [4] as they help create virtual power and virtual ground networks. The virtual power network drives the cells and can be turned off. The sleep transistors need to be tuned to a particular reference voltage by partially turning them on in idle mode.

As already mentioned, in our design we have dynamically included sleep transistor in series with power supply to take advantage of the stack leakage savings. A critical design component - SNM (signal noise margin) will be significantly lowered by sleep transistors. To achieve a particular SNM, the size of the sleep transistor is to be scaled linearly with respect to the number of cells in a column. As with any design, there are tradeoffs associated with sleep

transistors. Leakage current through sleep transistor is proportional to the width of the transistor. Small sleep transistors are more effective but they have negative impact on performance. When sleep transistors are upsized, leakage becomes less significant. Hence, sizing them is design dependent. In cases when delay could be tolerated, small sleep transistors are ideal for considerable power savings. In cases like Adders which are delay critical, large sleep transistors are optimal.

d) Design of the sleep transistor

Size is calculated using the average current method [5]. When average current is flowing through the sleep transistor and speed penalty for the SRAM block is known, the minimum size can be estimated. Gate delay time of a CMOS circuit is $\tau(V_{dd}) \alpha CV_{dd}/(\beta(V_{dd}-V_{tL})^{\alpha})$. Here $\beta$ is drivability factor, C is output load capacitance, $\alpha$ is saturation index and $V_{tL}$ is low $V_{th}$. MTCMOS Speed Penalty is $MSP=1/[1-(I_{sleep}/W_{sleep})-(R'/(V_{dd}-V_{tL})]^{\alpha-1}$ where R' is normalized sleep transistor resistance width of sleep transistor $W_{sleep}=(1/(1-MSP^{(1/1-\alpha)}))[R'/(V_{dd}-V_{tL})] I_{sleep.}$

The voltage swing is given by $\Delta v=I_{sleep}X R_{sleep}$ If $W_{sleep}$ is too large the current resonators suppress large ground forces and will not switch; if it is too small the ground bounce is up to 0.5 vdd. So typically the sleep transistor is 3% of SRAM cell area. Below is our SRAM cell with the sized sleep transistors.

e) Sleepy SRAM Cell – 10T:

Two pairs of Sleep transistors are used in the SRAM cell as shown. One in each pair is activated during idle mode based upon the value of the bit stored in the cell. This disconnects the OFF transistors from supply while retaining supply to the ON transistors [9].

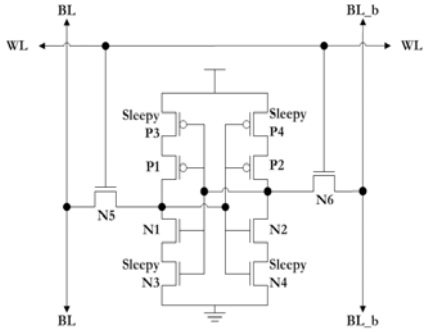


Fig 3. 10T SRAM Cell

| Transistor | W/L (nm) |
|---|---|
| N1 | 500/120 |
| N2 | 500/120 |
| N3 | 500/120 |
| N4 | 500/120 |
| N5 | 250/120 |
| P1 | 190/120 |
| P2 | 190/120 |
| P3 | 190/120 |
| P4 | 190/120 |
| N6 | 250/120 |

Table. II

III. 8K – bit SRAM

In this project, an 8K-bit SRAM is designed with 128 rows and 64 columns. Data is read out or written into memory in the form of 8-bit words. 7 address bits are required to decode a row and then 3 bits are required to access a word in a column. Each SRAM cell has a word line and two bit-charge lines

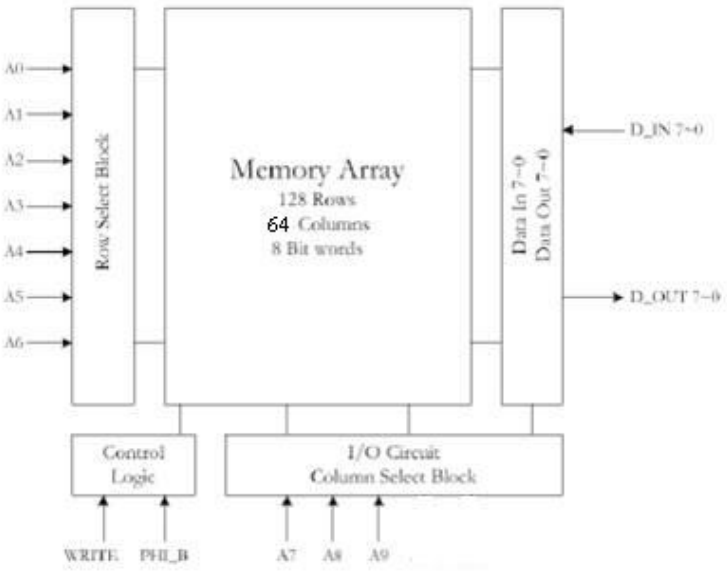


Fig 4. 8K-bit SRAM

a) Row decoder:

The Row Select block has 7 bits A0-A6 used to decode one of the 128 rows. This decoder has to select one of the word lines of 128 rows, each containing 64 1-bit SRAM cells. Decoder logic is designed to be fast enough so that it doesn't become the bottleneck of the whole design. For decoder logic we can either design with 2x4 decoders, 3–input AND gates or 3x8 decoders and 2-input AND gates. Considering space limitations on layout (3-input AND gates take much more area and offer high gate capacitance), it is designed with two 3x8 decoders and an inverter in the first stage. We AND the corresponding outputs in the second stage.

b) Column Multiplexer:

The column multiplexer is implemented as a 3x8 decoder with pass transistors connected to bit lines. Each output of column decoder is connected as an input to eight pass transistors of eight columns. This enables to read or write eight bits at a time. The 3x8 decoder is implemented with two input NAND and NOR gates.

c) Pre-charge circuitry:

The bit line conditioning circuitry is used to pre-charge the bit lines high before any operation. The bit lines need to be driven high before any data is written into the cell. This is accomplished by two PMOS transistors. The equalizing transistor connected between the BL and BL lines is used to reduce the pre charge time by ensuring that the two lines are at nearly equal voltages even if they are not charged all the way to $V_{dd}$. This is necessary to avoid differential noise on the bit lines which could cause the sense amplifier to fail.

d) Sense Amplifier:

A Sense Amplifier is an essential circuit in designing memory chips. Due to large arrays of SRAM cells, the resulting signal, in the event of a Read operation, has a much lower voltage swing. To compensate for that swing a sense amplifier is used to amplify voltage coming off BL and BL. It amplifies the difference of the signals on BL and BL signals to overcome signal distortions. It provides faster sensing by responding to a small voltage swing. The voltage coming out of the sense amplifier typically has a fully swing (0 – 1.8V) voltage. Sense amplifier also helps reduce the delay times and power dissipation in the overall SRAM chip.

e) Read – Write Circuitry:

RD, WR, DATA and PH (pre charge) are four inputs. When PH and RD are asserted high, Read circuit generates a sense clock to activate sense amplifier. When PH and WR are asserted high, Write circuitry will drive BL and BL_bar based on the data input.

## IV. Wire Delay Models

A memory cluster is a huge array and the word lines and bit lines will confront a huge wire load. The global interconnect wire in such designs have significant parasitic elements associated with them. The delay caused by these quasi elements must be taken into account in order to model accurately the behavior of the circuit and the power consumption of the circuit.

As we scale down the technology, the interconnect delay tends to exceed the device delay and is becoming the dominating factor in determining system performance. The interconnect delay for an average interconnect (1mm) in 130 nm technology was found to be 0.104ns [6]. We estimated the size of a single SRAM cell in our design and calculated the length of the word line and bit lines running parallel to the dimensions of the SRAM cell. We got an area estimate of 2.45 $\mu m^2$ with a horizontal dimension of 1.4 μm and a vertical dimension of 1.75 μm. The calculated dimensions concur with an Intel paper on 130nm technology based SRAM cell [7]. The 3 segment π- model is adopted to characterize the delay in our design as it estimates the wire characteristics within 3% error. The wire model not only includes the wire capacitance and resistance but also the gate and junction capacitance connected to the wire. The gate capacitance and drain capacitance for a single cell were found through HSPICE simulation to be around 0.1429fF/ $\mu m^2$ and 0.2697fF/μm. These values were then scaled up for the whole array and included in the wire model where each capacitance includes the gate for word lines and junction capacitance for bit lines. The importance of wire model is that it

determines the overall layout as sleep transistors can only be placed in cells that can tolerate the loads presented.

## V. Results

In this project, leakage power is measured by simulating a 1 Kb SRAM array as resources available to us restricted our choice further. Leakage power is measured when each SRAM cell holds a logical '1' or '0'. All simulations were done in HSPICE using 130 nm BPTM parameter file.

Leakage power of 10T SRAM cell (Sleepy) = 55.15 μW

Leakage power of 6T SRAM cell (non-sleepy) = 71.36μW

| Mode | Leakage Power | Reduction Rate |
|---|---|---|
| 100% sleepy | 19.36 mW | 47% |
| 75% sleepy | 23.28 mW | 37% |
| 50% sleepy | 27.58 mW | 25% |
| 25% sleepy | 31.9 mW | 13% |
| Non sleepy | 36.52mW | 0 |

Table. 3

The above table shows the leakage power and rate of reduction associated with the different kinds of implementation. Since, this is only for 1 Kb design, the reduction rate increases further as we move to higher density SRAM designs.

From the table, it can be seen that if we want to have maximum power reduction, then 100% sleepy mode is the best option. However, delay and area constraints associated with them make it less attractive. In the 1 Kb SRAM array, the read access time is measured as the time delay between the assertion of the address lines and the swing of the bit lines to $V_{DD}/2$. From 50% sleepy design to 100% sleepy the delay increases by 10% for a small design of 1 Kb. It would increase up to 30% with the increase in the size of the SRAM array as quoted in literatures related to memory designs. The above disadvantage could be overcome if leakage power is done by dynamically adjusting body bias along with sleep transistors to maintain performance comparable to conventional design.

## VI. References:


1. Weste and Harris "*CMOS VLSI Design- A circuits and systems perspective"*
2. D.Ho, K Iniewski, S. Kasnavi, A Ivanov, S. Natarajan-"*Ultra-Low power 90nm6T SRAM Cell for wireless sensor network applications"*
3. Kevin Zhang, Uddalak Bhattacharya et all- "*SRAM Design on 65-nm CMOS Technology with Dynamic Sleep Transistor for Leakage Reduction"*
4. Benton H. Calhoun, Frank A. Honore, Anantha P. Chandrakasan-"*A Leakage Reduction Methodology for Distributed MTCMOS"*
5. Mohab Anis, Mohamed Elmasry "*Multi-threshold CMOS Digital Circuits-Managing Leakage Power"*
6. Jason Cong *"Challenges and opportunities for design innovations in logic technologies"*-SRC Design Sciences concept paper
7. S. Tyagi, M. Alavi et all-"*A 130nm generation logic technology featuring 70nm transistors, Dual $V_t$ transistors and 6 layers of Cu interconnect"*
8. Anand Ramalingam, Bin Zhang, Anirudh Devgan, David Z. Pan- "*Sleep Transistor sizing using timing criticality and Temporal Currents"*
9. Zhengya Zhang, Zheng Guo *"Active Leakage Control with Sleep Transistors and Body Bias"*